

N94-36502

1994031995

Software Engineering Technology Transfer: Understanding the Process

Marvin V. Zelkowitz

Institute for Advanced Computer Studies
and Department of Computer Science
University of Maryland
College Park, Maryland 20742

Abstract

Technology transfer is of crucial concern to both government and industry today. In this report, the mechanisms developed by NASA to transfer technology are explored and the actual mechanisms used to transfer software development technologies are investigated. Time, cost, and effectiveness of software engineering technology transfer is reported.

"Technology transfer is a fundamental mission [of NASA]. It is as important as any NASA mission and it must be pursued."¹

Accordingly, NASA has set up several organizations within NASA, or affiliated with NASA, to deal with technology transfer. NASA has a perceived model of how technology transfer should operate. However, how well do these mechanisms actually work? What is the actual process used to transfer technology? What are the characteristics of technology transfer?

While NASA's main function is to develop space technology by building and launching satellites and manned missions, as a large technological organization, NASA must increasingly rely on computer technology to play an increasingly important role in all of its operations. Therefore, technology transfer of computer technology is also a major component of its technology transfer mission.

Therefore, given the existing model of technology transfer within NASA, how well does it address software technology? More specifically, given that:

1. Industry must transfer technology from developers to users,
2. Technology transfer is an integral part of NASA's mandate,
3. Software technology is an important component of many NASA activities,
4. Mechanisms have already been established by NASA to affect that transfer, and
5. NASA has a perceived model of this transfer.

we wish to learn:

¹Daniel S. Goldin, NASA Administrator, December, 1992.

1 Introduction

The transfer of technology from the developer to the consumer of that technology is of crucial concern to U. S. industry today as the need to remain economically competitive in a global marketplace forces all organizations to constantly improve their mechanisms for doing business. Government is not immune from these forces and needs to understand and participate in such activities at all levels.

The National Aeronautics and Space Administration (NASA), as a large government agency, plays a role as both a producer and consumer of such new technologies:

As producer. As the premier space agency of the United States, NASA has a mission to develop space technologies. Transferring these technologies to private industry and aiding in the commercialization of those technologies allows for government help in promoting U.S. industry internationally.

As consumer. However, with an annual budget of over \$15 billion, NASA is involved in a great many activities, and using the best techniques – whether developed internally or developed by those outside of NASA – enables NASA to wisely use its appropriated funds in order to work on complex tasks as economically as is practical.

NASA understands its role in technology transfer:

1. What is the real model used to affect technology transfer?
2. How well does this model work with software development technologies?
3. What software development technologies have actually been transferred successfully? and
4. What characteristics can we learn about technologies that have been transferred?

This report is organized as follows: In Section 2 some existing notions about technology transfer are presented and in Section 3 the current NASA model on technology transfer is given. Section 4 describes one general survey of industry that provides a rough baseline of technologies that have been transferred within the last 15 years, while in Section 5 NASA's role in both importing and exporting software engineering technologies are discussed. Conclusions from this work are given in Section 6.

2 Background

2.1 Process Improvement

Of great concern to all industry is the need to improve productivity. Within the computer science community, the ability to improve the process of developing software has been found to be a major impetus towards improving productivity and reliability of the resulting systems. Concepts like the Software Engineering Institute's Capability Maturity Model [2] have grown in importance as a means for modifying the software development process. The Experience Factory concept of the NASA/GSFC Software Engineering Laboratory (SEL) [1] has shown the value of process improvement.

However, all process improvement involves changes. Some of these may be relatively minor alterations to the current way of doing business (e.g. replacing one compiler or editor by another). However, some may require major changes that affect the entire development process (e.g., using cleanroom software development).

In order for an organization to continually improve its process, it must be aware of how it operates and what other technologies are available that may be of use. Understanding this process of technology transfer should enable NASA to better use its existing resources and to better plan for the future.

2.2 Technology transfer

When we discuss *technology transfer* we will mean the insertion of one technology into a new organization that previously did not use that technology. The insertion must be such that the new organization regularly uses that technology if the appropriate conditions on its use should arise in the future.

We will call the original creator of that technology the *producer* of the technology and the organization that accepts and uses the new technology the *consumer* of that technology. The process of moving the technology out of the producing organization will be called *exporting* the technology while the process of installing the technology in the new organization will be called *infusing* the technology.

Implied by the above definitions is the notion that a successfully transferred technology becomes part of the state-of-the-practice, or normal operating procedures, of the infusing organization. For example, an organization that experiments with Ada as a programming language and then decides to use it for all applications in a specific domain (e.g., for all flight simulators) can be said to have successfully transferred that technology. On the other hand, if a technology is tried once or twice (e.g., the ML programming language for expert system development) and is found wanting and will not be used again, then that technology will not be considered to be transferred.

Not transferring a technology does not imply that the technology is not effective; only that it does not apply to the particular consumer domain. For example, there is still a demand for buggy whips among horse enthusiasts and certain theme park operators, but they have few applications among most urban automobile repair shops.

What technology are we interested in?

"Technology" is a very imprecise concept. For this report we are mainly concerned with tools, procedures and mechanisms that aid in the development of software products. We can divide this domain into two categories:

Software development technology. This includes the tools and procedures used by the software engineering profession to build software. It includes, in addition to the usual computer-based items like machines, editors, compilers, testing tools and configuration management systems, items like electronic mail, desktop publishing, spreadsheets and any other tool or device useful for software production. This can even include the telephone or fax machine if either provides aid in the development of software.

	NASA Consumers	External Consumers
NASA Producers	Transferred within NASA	Exported from NASA
External Producers	Infused into NASA	Not of interest

Table 1: Participants in technology transfer

Software engineering technology. This includes those software development technology items created specifically for software development. Thus, while it will include compilers and testing tools, it will not include items like electronic mail or the fax machine which also have uses in other domains.

2.3 Technology transfer participants

In describing the transfer of technology into and out of NASA, we have four potential groups of producers and consumers to consider (Table 1). NASA may be either a producer of a consumer of some technology. Similarly, some other organization may be either the producer or consumer of that technology. Of the four potential cases, only three are considered in this report – those involving NASA as either a producer or consumer. The case where an external producer transfers a technology to an external consumer is certainly of interest, but is outside of the scope of this work on NASA's role in technology transfer.

2.4 Technology maturation

In 1985, Redwine and Riddle [3] published the first comprehensive study of software engineering technology maturation. Their goal was to understand the nature of technology maturation – what was the length of time required for a new concept to move from being a laboratory curiosity to general acceptance by industry. They defined maturation of a technology as a 70% usage level across the industry.

Technology maturation involves 5 stages – two by the producer of the technology and three by consumers of that technology (See Figure 1):

1. The original *concept* for the technology appears as a published paper or initial prototype implementation. The initial time period is the development of the concept by the originator of the technology.

2. The *implementation* of the technology involves the further development of the concept by the originator or successor organization until a stable useful version is created.

3. In the initial *experimental* (or understanding) stage, other organizations experiment, tailor, expand, modify and try to use the technology.

4. In the later *exploration* (or transition) stage, use of the technology is further modified and expands penetration across the industry.

5. The final *maturation* stage is reached when 70% of the industry uses the technology.

In their study, they looked at 17 software development technologies that were developed from the 1960s through the early 1980s (e.g., UNIX, spreadsheets, object oriented design, etc.). Their results, most related to this current project are:

- They were unable to clearly define "maturation" for most technologies, but were able to make reasonable estimates as to the length of time needed for new technologies to be widely available.
- Technologies required an average of 17 years to pass from an initial concept to a mature product.
- Technologies, once developed, required an average of 7.5 years to become widely available.

In this current study, we are not interested in the general issue of technology maturation, but instead the infusion (or exporting) of a technology into or out of a single organization (NASA). Therefore, we would expect this 7.5 year average exploration stage to be an upper bound. What would be a reasonable value for NASA-infused or developed software technology?

3 NASA Model of Technology Transfer

Since technology transfer is part of NASA's mandate, a model of technology transfer has grown within the agency. Several offices and related organizations have been created for dealing with technology transfer. These include the following organizations:

- Technology transfer organizations at each NASA center:

Technology Utilization Office. The Technology Utilization Office (TUO – or Technology Transfer Office (TTO) as part of Code 700 (Engineering) at GSFC) is the major proponent of technology

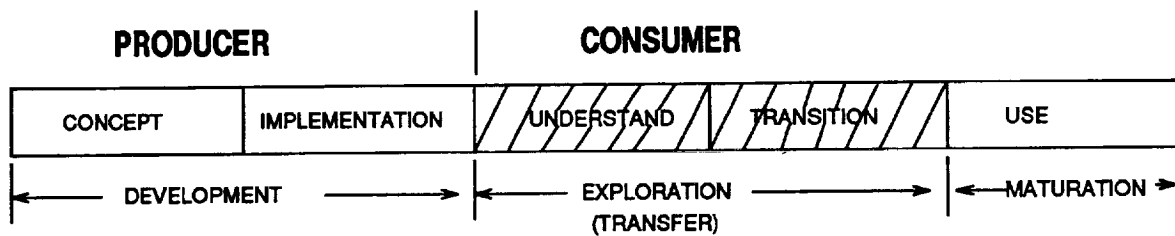


Figure 1: Technology Maturation Life Cycle

transfer between the engineer and industry. Its focus is to aid the NASA engineer in moving an idea into industry.

Office of Commercial Programs. The Office of Commercial Programs (OCP) is the major interface between each NASA center and industry as an intermediary in the commercialization of concepts that arose in NASA research.

- National and regional technology transfer organizations:

National Technology Transfer Network. The National Technology Transfer Network and the various regional technology transfer field centers act as intermediaries between the individual TUOs at each center and industry. Six Regional Technology Transfer Centers (RTTC) work directly with industry to aid in the commercialization of NASA products.

Technology Application Team. The Technology Application Team (TAT) is located at Research Triangle Institute and works with each TUO in developing technology transfer projects.

COSMIC. The Software Technology Transfer Center is a repository of software developed by NASA personnel. It is located at the University of Georgia. Over 5,000 programs have been submitted to COSMIC for distribution since 1966.

- Technology transfer publications and agreements:

Space Act Agreements. Space Act Agreements (SAA) are like memoranda of Understandings (MOUs) or CRADAs (Cooperative Research and Development Agreements in other federal agencies) for joint industry–NASA cooperation on specific projects.

NASA Tech Briefs. “NASA Tech Briefs” is a monthly publication for announcing new inventions and innovations.

	NASA Consumers	External Consumers
NASA Producers	COSMIC Tech Briefs Conferences Papers	COSMIC Tech Briefs Conferences Papers TAT NTTN TUO OCP SAA Spinoff
External Producers	Conferences Papers	<i>Not of Interest</i>

Table 2: NASA Tech Transfer Model

Spinoff. “Spinoff” is an annual publication that summarizes those technologies that have been successfully commercialized during the previous year.

Conferences and publications. Conferences and publications (both NASA sponsored and non-NASA sponsored) are a major source of information on technology that has been produced both within and outside of NASA.

Merging this list of technology transfer mechanisms with our previous model of technology transfer participants (Table 1), we get a clearer picture of how NASA addresses technology transfer (Table 2).

Two results become immediately apparent from this table:

1. There is no infusion mechanism for bringing new technology into the agency.
2. The major goal is the transfer of products.

The former result, under today's economic climate, needs to be reevaluated. Previously, there was a desire on the part of Congress and those in charge of NASA to show that such large sums of money spent on space applications had practical benefits for U.S. industry. Therefore, technology transfer to industry gave concrete indications of the value of space exploration.

However, the situation today is an era of static or shrinking budgets. The concept of "Faster, Better, Cheaper" is heard more and more. NASA needs to "Work Smarter." One way to do that is to use better technology and infuse better techniques into the agency. However, the current model assumes that engineers can simply learn about such technologies from reading papers and going to conferences. There is no explicit aid to help in this search for better ways to do NASA's job.

The second result, transfer of products, also needs to be reevaluated in the light of software engineering technologies. In most engineering disciplines, processes are centered in various products that implement that technology. Thus transferring a technology is generally equivalent to transferring a product.

The same cannot be said of many software engineering processes. For example, within the GSFC Software Engineering Laboratory (SEL), the following list of processes have been studied over the past few years:

- Object Oriented Technology,
- Goals/Questions/Metrics paradigm of software development,
- The Experience Factory model of development, and
- Cleanroom software development.

None of these processes is embodied in a product. One cannot buy a "Cleanroom" program. Instead one buys some books, a training course and some guidance on using the technique. Although NASA does not explicitly address the packaging of such processes as assets to be transferred, NASA is not unique in this regard. It is not clear that much of industry understands the unique role that processes play in software development compared to most engineering processes. It is imperative for NASA to understand this distinction and to address the transfer of processes as well as products.

Total Replies (44)		NASA Replies (12)	
Item	No.	Item	No.
Workstations & PCs	27	*Object oriented	12
*Object oriented	21	Networks	10
GUIs	17	Workstations & PCs	8
*Process models	16	*Process models	7
Networks	16	*Measurement	5
*C and C++	8	GUIs	4
*CASE tools	8	*Structured design	3
Database systems	8	Database systems	2
Desktop publishing	8	Desktop publishing	2
*Inspections	7	*Dev. methods	2
Electronic mail	7	*Reuse	2
		*Cost estimation	2
		Comm. Software	2

* – Software engineering technologies

Table 3: Top 10 transferred technologies

4 Software Development Technology

In order to understand software engineering technology transfer within NASA, it was first necessary to understand if there was any consensus about how software development technology has evolved over the past decade. Papers are often written about the so-called "software crisis" with comments that software development has not changed at all in 30 years. If so, then obviously no technology has been transferred lately.

In order to address this, a brief survey form was prepared and widely distributed. The form asked for the top five technologies that have changed software development practices since 1980. A list of approximately 100 items was included, and the respondent could pick from that list or add any other items that seemed relevant.

A total of 44 forms were returned. Of these, 12 were from NASA/GSFC personnel or NASA contractors. The top 10 (including ties) from the total list and the NASA list are given in Table 3.

The level of consensus among the 44 returned forms was quite surprising. The top 5 in the list clearly dominated the others. Of the top 5, only two of them (object oriented technology and process models) were clearly software oriented technology. Two were mostly hardware (workstations and networks) and the other (graphical user interfaces) was partially a software engineering issue, but does not strictly fit into our definition of software engineering technology given its use in many application programs.

Among the NASA replies, there was likewise strong

	NASA Consumers	External Consumers
NASA Producers	Reuse(Kaptur, CLIPS) Environments (SSE) GUI(TAE) Measurement(SME, GQM) AI tools CASE Tools	Reuse(Kaptur, CLIPS) Environments (SSE) GUI(TAE) Measurement(SME, GQM)
External Producers	Cost models * Formal Inspections * Object oriented technology * Ada, C, C++ * Cleanroom Rate monotonic scheduling CASE tools	<i>Not of Interest</i>

* – technologies discussed in more detail

Table 4: Transferred technologies at NASA

agreement with the total list. The top 5 on the total list were 5 out of 6 among the NASA entries. Only measurement, a strong component of the SEL, raised the recognition level among NASA personnel of its importance.

Other technologies that have been claimed as effective that were mentioned in the survey include: Measurement (in 12th place), Ada (in 14th), Formal methods (in 17th) and UNIX (in 17th).

Clearly some software development technologies have been transferred. The question was now: What effect did NASA have on this transfer?

5 Software Engineering Technology

In order to understand technology transfer in NASA, three or four software engineering experts at each NASA center were interviewed in order to determine what software engineering techniques were being used effectively in the agency. In order to keep the scope of this problem reasonable, the following two restrictions were imposed:

1. The technology had to clearly be software engineering. For example, successfully transferred programs, such as the modelling system NAS-TRAN available through COSMIC, were not included.
2. The technology had to have a major impact on several groups within NASA. With more than 4,000 software professionals affiliated with GSFC

alone (including government and contractors), almost every software product has probably been used somewhere in the agency. While this was somewhat subjective, a list of transferred technologies was developed (Table 4).

Those technologies used (i.e., consumed) by NASA and those technologies produced by NASA will be considered separately.

5.1 Technology Infusion

Within the SEL

Three technologies that were successfully transferred by the SEL (Ada, Object oriented technology, and Cleanroom) will be discussed in greater detail. The details of transferring those technologies are summarized by Figure 2 and are explained in greater detail in the following subsections.

SEL use of Ada

Understanding phase of Technology Transfer.

Use of Ada on SEL projects was first considered in 1985. A training and sample program was the first Ada activity. However, in order to truly evaluate the appropriateness of Ada within the SEL environment, a parallel development of an Ada and FORTRAN simulator (GRODY) was undertaken. The results was an operational, but slow, product. Although the development of this simulator continued until early 1988, by early 1987 it was decided that the initial project was sufficiently successful to continue the investigation of

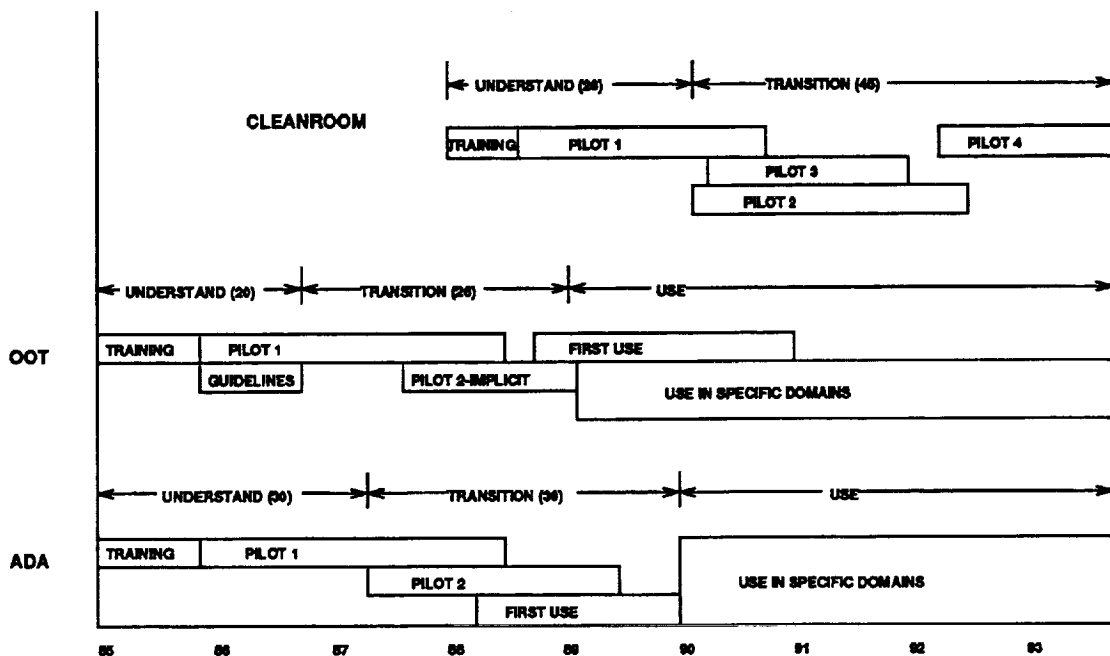


Figure 2: SEL technology transfer experience

Ada. Elapsed time since start of Ada activity was 30 months.

Transition phase of technology Transfer.

Because of the poor performance on the GRODY simulator, a second Ada project (GOADA) was undertaken where performance was of greater concern. In this case, the resulting product was comparable to performance of previous FORTAN simulators. In 1988 a third simulator was undertaken and developed successfully. By the end of 1989, Ada became the language of choice for simulators in the flight dynamics branch. Transition time was another 30 months.

Comments on Technology Transfer.

Total transfer time for Ada was approximately 60 months. Ada is the language of choice for simulator projects. Although Ada code costs more, line by line, than FORTRAN code, the higher levels of reuse result in lower overall delivery costs for such projects.

Ada was also used as the implementation language on larger mission ground support software. This was not as successful. However, the inhibitors in this case were outside of the scope of the Ada language, itself. The operational systems at GSFC are IBM mainframe compatible, and no effective Ada compiler existed for this environment during the 3 times Ada was evaluated. All of the successful simulator projects were implemented on DEC VAX computers, which had an

effective Ada system.

Presently, Ada is used on approximately 15% of the SEL's software. Eleven Ada projects have been completed to date.

SEL use of object oriented technology

Understanding phase of Technology Transfer.

Use of object oriented technology (OOT) in the SEL was seriously investigated at the same time as Ada was considered. In developing the requirements for GRODY, it became apparent that the standard GSFC requirements document was oriented towards a FORTRAN functional decomposition and the use of these requirements on an Ada project would be very inefficient.

Therefore the requirements were redone to use a more object oriented approach. Following this, an OOT guidebook for GSFC was developed (GOOD - General Object Oriented Software Development [4]) for use on future projects.

Elapsed time for these activities took from early 1985 until August, 1986, or a total of 20 months. Expenses for understanding this technology were high since this activity was wrapped up in the Ada evaluation which required parallel system development.

Transition phase of technology Transfer.

On a second project (UARSAGSS), object oriented

design was used implicitly. This was a FORTRAN ground support system, and experiences gained from the earlier GRODY effort allowed the programmers to better understand the design and use OOT. By the end of this project, it was sufficiently clear that OOT was an effective technique in some domains. Transition time was on the order to 26 months.

Comments on Technology Transfer.

Total transfer time in this case was only 46 months. Although almost 4 years, this was relatively short since it did not require major changes in system development. OOT could be used with Ada, FORTRAN, or any other programming language. Since it fit within the usual development paradigm, tailoring the method and inserting it into the usual NASA development process was relatively easy.

SEL use of cleanroom

Understanding phase of Technology Transfer.

In order to understand cleanroom, a series of training courses were given in 1988 by Dr. Harlan Mills, original developer of the method. A pilot project was undertaken and proved to be very successful. All participants were converts to the method, even though several had reservations about it before they began. Time to understand the method (training until the start of the second cleanroom project) was 26 months.

Transition phase of technology Transfer.

Two follow-on cleanroom projects were undertaken. A smaller in-house development was very successful, but a larger contracted project was not so successful. It was not as apparent that problems on the larger project were due to scaling up of cleanroom to larger tasks or to a lack of training and motivation of the development team on this project.

Because of this, a fourth cleanroom project was undertaken. This project is still under development, but preliminary results look very promising.

Comments on Technology Transfer.

Cleanroom technology appears to be an effective technology. Understand time was 26 months and transition time is at least 46 months, with transition still underway. Cleanroom cannot be said to be a mature technology yet, although results look very good.

Technology infusion at other NASA centers

Use of Formal Inspections

Transfer of technology.

Work began by John Kelly on formal inspections at JPL. The elapsed time for developing the method

was 30 months and involved about 6 staff months of effort. Contacts with Mike Fagan of IBM, developer of the technique, aided the transition process. It has been successfully transferred to JPL and between its initial use in February, 1987 through 1990, over 200 inspections were carried out.

Based upon experience at JPL, formal inspections were moved to Langley. This took only 16 months, because of the previous experiences at JPL. About 12 staff months of effort were required, but most of this effort was in "unpaid overtime." No NASA support was available for developing the technology.

Once installed at Langley, it has been transferred to several contractors working at Langley.

Comments on Technology Transfer.

Formal inspections were successfully transferred at Langley. Total time for transferring at both centers were 30 months and 16 months. These were relatively short since formal inspections cover only a relatively narrow and precise process in software development and can be inserted relatively easily into almost any mature development process.

5.2 Exporting Technology

This present version of this report is concerned mainly with the infusion of technology from outside of the agency. A later version will address technology exporting in more detail.

6 Conclusions

From this initial study, we can make several conclusions and observations:

1. NASA mechanisms do not address software engineering technologies well. Technology infusion is generally ignored and left up to the individual engineer to discover on his own what is needed and available. With today's shrinking budgets and the need to work "Better, Faster, Cheaper," NASA needs to address this issue and help in the search for new effective technology to use.

In addition, software engineering processes are not addressed. These are not product centered. How to package and transfer processes as a corporate asset needs to be handled better.

2. Technology transfer is more than simply understanding the new technology. Technology transfer takes time. Understanding the technology has been shown to take upwards of 2.5

years. The transition time when the organization is exploring, tailoring and modifying the technology for its own use often takes more than the understanding time, with a total transfer time on the order of five years not being unusual.

3. Technology transfer is part of the total environment of the consumer organization. Technology transfer does not occur in a vacuum. The SEL experience with Ada demonstrates this concept. Ada has proven to be successful with flight simulators, and the effective Ada compiler on the DEC VAX computer helped in this transfer. However, the operational systems for flight dynamic software was the IBM mainframe, and no effective Ada compiler was available during the 5 years (from 1985 to 1990) when Ada was under evaluation. Because of this, FORTRAN is still the language of choice for such applications. Had an effective mainframe Ada compiler been available, then the result of evaluating Ada for large AGSS (Attitude Ground Support Software) systems might have been different.

4. People contact is the main transfer agent of change. As many others have observed, technology transfer occurs best when the developers of a technology are involved in the technology transfer process. In this report, that happened in order for cleanroom to be effectively used at GSFC and for inspections to be brought to JPL and then to Langley.

7 Acknowledgment

This research supported in part by grant NSG-5123 from NASA Goddard Space Flight Center to the University of Maryland, College Park, Maryland. Frank McGarry of NASA/GSFC and Kellyann Jeletic of Computer Sciences Corporation helped in collecting the data described in this report.

References

- [1] Basili V. R., The experience factory: Can it make you a 5?, 17th NASA/GSFC Software Engineering Workshop, Greenbelt, MD (December, 1992) 55-64.
- [2] Paulk M. C, B. Curtis, M. B. Chrissis, and C. V. Weber, Capability Maturity Model for Software, Version 1.1, Technical Report SEI-93-TR-24, Software Engineering Institute, Pittsburgh, PA (1993).
- [3] Redwine S. and W. Riddle, Software technology maturation, 8th IEEE/ACM International Confer-

ence on Software Engineering, London, UK, August, 1985, 189-200.

- [4] Seidewitz E. and M. Stark, General Object Oriented Software Development, SEL-86-002, NASA/GSFC, August, 1986.



1993 Software Engineering Workshop

Software Engineering Technology Transfer:

Understanding the Process

Marvin V. Zelkowitz

Institute for Advanced Computer Studies and

Department of Computer Science

University of Maryland

College Park, Maryland



Technology Transfer Overview

- . **"Technology transfer is a fundamental mission. It is as important as any NASA mission and it must be pursued." – Daniel S. Goldin, NASA Administrator, December, 1992**
- . **It is critical to understand technology transfer as part of any process improvement program.**
- . **Technology maturation takes time: From Redwine – Riddle study (1985) on software engineering technologies:**
 - **Studied 17 software engineering technologies of the 1960s and 1970s.**
 - **Required an average of 17 years from concept to maturation.**
 - **Required an average of 7.5 years after initial development to widespread availability in industry.**
- . **Fundamental issues:**
 - How does NASA think technology transfer takes place?**
 - How does technology transfer really take place?**



Technology Transfer Goals

. Issues:

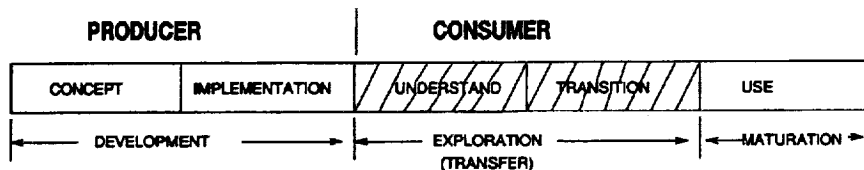
- Must transfer technology from developers to users
- Technology transfer is an integral part of NASA's mandate
- Mechanisms have been established by NASA for transfer
- NASA has a perceived model of this transfer

. But:

- What is the real model of technology transfer?
- What processes were used to affect those transfers?
- What software development technologies have been transferred?
- What were the costs and effort for those transfers?



Technology Transfer Stages



This initial study covers technology infusion only (e.g., Exploration stage within NASA.)



Domain of Interest

. What technologies are of interest?

- **Software development technology** – Tools and procedures used by software engineering profession to build software
- **Software engineering technology** – Tools and procedures developed specifically for software development

. What technologies do software engineers use?

- Software development technologies in use – Present broad-based survey of software engineering professionals on what software development technologies have been successful since 1980.
- Software engineering technologies transferred by NASA – Present results of interviews and surveys with selected NASA personnel and reading selected NASA documentation and reports.



Technology Transfer Participants

	NASA Consumers	External Consumers
NASA Producers	Transferred within NASA	Exported from NASA
External Producers	Infused into NASA	<i>Not of interest</i>

Purpose of this analysis is to understand both the exporting and importing of software engineering technology for NASA



Technology Transfer Parameters

• From Producers:

- Motivation (need) for technology
- Cost of technology
- Time to develop technology
- Commercialization potential of technology
- Cost and time to transfer technology

• From Consumers:

- Motivation (need) of technology
- Methods to investigate technology
- Cost required to infuse technology
- Time required to infuse technology



Top 10 Recently Transferred Technologies

Total Replies (44)		NASA Replies (12)	
Item	No.	Item	No.
Workstations and PCs	27	*Object oriented	12
*Object oriented	21	Networks	10
GUIs	17	Workstations and PCs	8
*Process models	16	*Process models	7
Networks	16	*Measurement	5
*C and C++	8	GUIs	4
*CASE tools	8	*Structured design	3
Database systems	8	Database systems	2
Desktop publishing	8	Desktop publishing	2
*Inspections	7	*Development methods	2
Electronic mail	7	*Reuse	2
		*Cost estimation	2
		Comm. Software	2
*Measurement (12)	6	*C (14)	1
*Ada (14)	4	*Ada (14)	1
*Formal methods (17)	3	*Inspections (14)	1
UNIX (17)	3		

* – Software engineering technologies



NASA Transfer Mechanisms

- . NTTN – National Technology Transfer Network. Joint NASA and other Federal agency transfer centers. NASA field centers and regional technology transfer centers for interacting with industry.
- . COSMIC – NASA Software Technology Transfer Center. At University of Georgia to make NASA software available.
- . TAT – Technology Application Team. At Research Triangle Institute, works with each Technology Utilization Office at each NASA center for in developing technology transfer projects.
- . Space Act Agreement – Joint NASA/industry project. (Similar to MOUs, CRADAs)
- . TUO – Technology Utilization Office – Office at each center for interacting with outside agencies
- . NASA Tech Briefs – Monthly publication for announcing new inventions and innovations.
- . Spinoff – Annual NASA publication describing successfully transferred technologies.



NASA Transfer Model

	NASA Consumers	External Consumers
NASA Producers	COSMIC Tech Briefs Conferences Papers	COSMIC Tech Briefs Conferences Papers TAT NTTN TUO Space Act Agreements Spinoff
External Producers	Conferences Papers	<i>Not of Interest</i>

Goal is transfer of products.

No infusion mechanism.



Software Engineering Processes

- . **Technology Transfer is generally product oriented – In most engineering disciplines, the process is centered in the product.**
- . **Software engineering does not yet fulfill that model – Processes describing actions to take are as important as the tools that are used.**
- . **For example, many of the technologies explored by the GSFC Software Engineering Laboratory are procedures only and not tools:**
 - **Object oriented technology**
 - **Goals/Question/Metrics model**
 - **Measurement**
 - **Cleanroom**
 - **Inspections**

Processes as opposed to products are dominant.



NASA Emphasis on Technology Transfer

- . **Summary of NASA Technology Transfer Model:**
 - **Agents of technology transfer are people.**
 - **Description of technology transfer are published papers.**
 - **Objects of technology transfer are products.**
- . **But:**
 - **No mechanisms for transfer of processes.**
 - **Seems to be true throughout industry, not just NASA.**
 - **No mechanism for technology infusion.**



What Has Been Transferred?

- . Domain of interest – Software engineering technologies (e.g., Most programs in COSMIC are application programs and not of interest for this talk.)
- . But NASA is big ...
 - Thousands of programmers nationwide. Probably every tool sold has been used somewhere within NASA.
 - Need to identify only those technologies that have made major impact on development practices
- . Preliminary results of directed survey of software engineering professionals within NASA.



Transferred Software Development Technology

	NASA Consumers	External Consumers
NASA Producers	Reuse(Kaptur, CLIPS) Environments (SSE) GUI(TAE) Measurement(SME, GQM) AI tools CASE Tools	Reuse(Kaptur, CLIPS) Environments (SSE) GUI(TAE) Measurement(SME, GQM)
External Producers	Cost models * Formal Inspections * Object oriented technology * Ada, C, C++ * Cleanroom Rate monotonic scheduling CASE tools	Not of Interest

* – Technologies to be discussed

Representative list based upon survey and interviews



Case study: SEL use of Ada

Understand time 30 mo
Transition time 30 mo
Cost High, Parallel development
Replaces FORTRAN
Infusion method Courses, 2 pilot projects
Status Mature use in specific domain
Tech Transfer Used on some projects
Success Generally positive

Comments

- . Increased costs for new projects
- . 10%–25% savings on later projects due to 25% – 30% reuse



Case study: SEL use of OOT

Understand time 20 mo
Transition time 26 mo
Cost High (Part of Ada evaluation)
Replaces Functional decomposition
Infusion method Courses, Training guide, 2 pilot projects
Tech Transfer Used on most projects
Status Mature use in specific domains
Success Very positive

Comments

- . Initial results – Decreased time and effort and improved error rates
- . Needs training – Replaces design method that already worked well and generates few errors



Case study: SEL use of Cleanroom

Understand time 26 mo

Transition time 45+ mo

Cost High

Replaces Traditional testing

Infusion method External developers, training, pilot studies

Status Still in transition

Tech Transfer Unclear

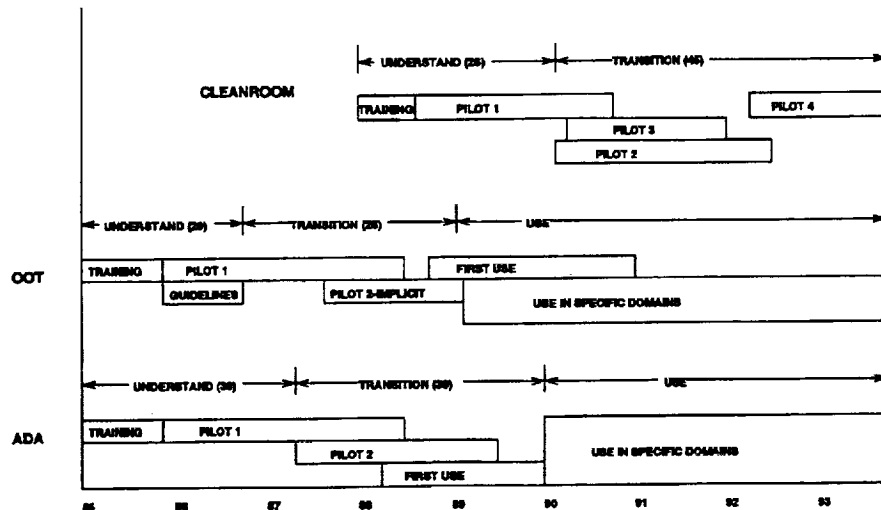
Success Appears very positive

Comments

- . Contact with developer important for early success
- . Large project not as successful – less training and motivation
- . Productivity and error rates improved on all projects
- . Still evaluating, training and undergoing transition



Summary of SEL Experiences





Case study: Formal Inspections

	Site 1	Site 2
Transfer time	30 mo	16 mo
Cost	.5 FTE	1 FTE, unpaid overtime
Replaces	Walkthroughs	New activity
Infusion method	External developer	External developer, site 1
Status	In use	In use
Tech Transfer	Used within NASA, site 2	NASA contractors government
Success	High	High

Comments

- . Technology transfer not well supported
- . People contact main agent of change



Conclusions

NASA mechanisms do not address software engineering technologies well.

- . Technology infusion is generally not addressed.
- . Process technology is similarly not addressed.

Technology transfer is more than simply understanding the new technology.

- . Time to understand technology is generally on the order of 2.5 years.
- . Transition time at least as long as understanding time.

People contact seems to be the main transfer agent of change.

Disclaimer: Presentation based upon preliminary analysis of available information. Will be refined over next few months.